ISSN : 2548-6985

Penerapan Steganografi Video Dengan Metode Discrete Cosine Transform

Abdi Ansor
STMIK Budi Darma Medan, Jl. Sisingamangaraja No. 338 Medan, Sumatera Utara, Indonesia
E-Mail : abdiansor@gmail.com

ABSTRAK

Teknik-teknik penyisipan pesan ke dalam video dapat dikategorikan berdasarkan dua domain, yaitu domain spasial dan domain frekuensi. Penyisipan pada domain spasial dilakukan dengan menyisipkan langsung ke dalam bagian yang tampak pada frame video, seperti pada komposisi kandungan warnanya, atau pada kandungan luminancenya. Walaupun proses ini akan mengubah tampilan dari video, namun proses penyisipan ini dapat dilakukan sedemikian rupa sehingga perubahan yang diakibatkan tidak dapat dipersepsi oleh mata manusia. Sedangkan pada domain frekuensi, penyisipan dilakukan ke dalam hasil dari proses transformasi frame video ke dalam frekuensi. Penyisipan ini tidak mengakibatkan perubahan pada bagian yang tampak, namun proses perhitungannya membutuhkan kalkulasi yang lebih rumit dari pada penyisipan pada domain spasial.
DCT *(Discrete Cosine Transform*), yang merupakan salah satu metode untuk penyisipan pesan. DCT adalah proses untuk mengubah domain spasial gambar menjadi domain frekuensi, dan digunakan pada gambar berformat JPEG. DCT akan mengubah koefisien DCT dan menyisipkan dalam koefisien tersebut. DCT disebabkan penurunan kualitas pada video yang dihasilkan tidak signitifikan, dan pesan di dalamnya tidak akan hilang apabila dilakukan perubahan terhadap terhadap video tersebut.

Kata Kunci : Steganografi, Video, Pesan, DCT

PENDAHULUAN

Kebutuhan untuk mengirimkan informasi dari suatu tempat ketempat lain menjadi sangat mudah untuk dilakukan pada masa sekarang ini. Teknologi perangkat keras yang digunakan mengalami perkembangan yang pesat, begitu juga dengan kompleksitas algoritma dari perangkat lunak yang digunakan didalamnya. Hal ini menyebabkan proses pengiriman informasi menjadi cepat. Jenis informasi yang bisa dikirimkan juga semakin beraneka ragam, dari hanya mengirimkan tulisan yang berukuran kecil, hingga bentuk multimedia yang membutuhkan perhitungan rumit seperti *video*.

Media yang sering menjadi tempat penyembunyian pesan pada *steganografi* digital adalah teks, gambar, suara, dan *video.* Pemilihan jenis media yang digunakan sembarang, tetapi apabila pesan yang ingin disembunyikan berukuran besar, maka bentuk *video* merupakan format yang cocok sebagai medianya. Menurut jurnal Muhammaddin Yunus program studi ilmu komputer , FMIPA UGM Vol.8,No1, Januari 2014 ISSN: 1978-1520. “Penyembunyian data pada *file video* dengan metode LSB dan DCT”. Format *video* mengandung sejumlah *frame* dimana masing-masing *frame* tersebut dapat disisipi sebagaimana halnya dengan penyisipan yang terjadi pada gambar, sehingga kapasitas penyimpanan pada *video* menjadi besar. Teknik-teknik penyisipan pesan ke dalam *video* dapat dikategorikan berdasarkan dua domain, yaitu domain spasial dan domain frekuensi. Penyisipan pada domain spasial dilakukan dengan menyisipkan langsung ke dalam bagian yang tampak pada *frame video*, seperti pada komposisi kandungan warnanya, atau pada kandungan *luminance*nya. Walaupun proses ini akan mengubah tampilan dari *video*, namun proses penyisipan ini dapat dilakukan sedemikian rupa sehingga perubahan yang diakibatkan tidak dapat dipersepsi oleh mata manusia. Sedangkan pada domain frekuensi, penyisipan dilakukan ke dalam hasil dari proses transformasi *frame video* ke dalam frekuensi. Penyisipan ini tidak mengakibatkan perubahan pada bagian yang tampak, namun proses perhitungannya membutuhkan kalkulasi yang lebih rumit dari pada penyisipan pada domain spasial[1].

Metode yang digunakan penyisipan pesan dalam penelitian ini adalah metode DCTM *(Discrete Cosine Transform)*, yang merupakan salah satu metode untuk penyisipan pesan pada domain frekuensi. DCT adalah proses untuk mengubah domain spasial gambar menjadi domain frekuensi, dan digunakan pada gambar berformat JPEG. DCT akan mengubah koefisien DCT dan menyisipkan dalam koefisien tersebut. Penggunaan DCT disebabkan penurunan kualitas pada *video* yang dihasilkan tidak signitifikan, dan pesan di dalamnya tidak akan hilang apabila dilakukan perubahan terhadap terhadap *video* tersebut[2].

ISSN : 2548-6985

Berdasarkan latar belakang di atas maka perumusan masalah adalah sebagai berikut :

1. Bagaimana proses ekstraksi *file video* ke dalam bentuk *frame*?
2. Bagaimana menyisipkan pesan terhadap *video* menggunakan metode DCT (*Discrete Cosine Transform*)?
3. Bagaimana merancang aplikasi steganografi *video* ke dalam *software Visual Basic*.Net 2008 ?

Untuk batasan masalah dari permasalahan di atas adalah sebagai berikut : *Video* digital yang digunakan adalah berformat 3GP *grayscale* (hitam putih) dengan size KB., *Video* yang digunakan tanpa *audio.*Pesan yang disisipkan adalah 8 bit, dalam bentuk teks dan jumlah karakter tidak melebihi penyimpanan pada *file video,* Pengambilan *frame file video* menggunakan *convert video to jpeg,* Penggabungan *frame file video* menggunakan *ulead.,* Metode yang digunakan DCT yang 1-D., Tidak membahas pengambilan pesan dari *video*.

Adapun tujuan penelitian ini adalah Untuk mengestraksikan *file video* ke dalam bentuk *frame*. Dalam penyisipan pesan dengan *video* menggunakan metode DCT *(Discrete Cosine Transform).* Merancang aplikasi steganografi *video* menggunakan metode DCT *(Discrete Cosine Transform).*

Sedangkan manfaat penelitian ini adalah :

1. Dapat mengamankan pesan rahasia sehingga aman dari orang-orang yang tidak berkepentingan yang berusaha untuk mengetahui.
2. Sebagai referensi untuk materi yang masih berhubungan untuk dikembangkan lebih lanjut di kemudian hari.
3. Agar dapat dijadikan menjadi sebuah aplikasi *alternative* dalam penyembunyian pesan yang aman pada *file video*.

Sebuah *video* digital terdiri dari *frame-frame* yang mana *frame-frame* tersebut dikompres menjadi sebuah *file* komputer yang hanya dapat dijalankan menggunakan sebuah perangkat lunak multimedia player.

Berdasarkan bentuk-bentuk kompresan dari *file video* digital tersebut, banyak bermunculan format-format *video* digital yang ditawarkan kepada pengguna dengan kelebihan dan kekurangannya masing-masing. Adapun beberapa contoh dari format *video digital* yang sering dijumpai antara lain :

1. AVI (*Audio Video Interleave)*
   AVI merupakan format *video* dan animasi yang digunakan *windows* dan berekstensi AVI. Sebagian besar *authoring* pada *windows* mendukung format AVI juga didukung oleh *Netscape*. Kekurangan penggunaan *AVI* pada *playback* adalah pemakaian *Macintosh*, SGI dan *Sun* harus mengubah *file* ke format lain untuk *playback*. Format AVI memang masih kurang canggih, berbasis *track* dan kemampuan dalam melakukan *sinkronisasi* dengan *Quick Time* kurang bagus, *codec* untuk *Quick Time* pada *windows* lebih berkembang daripada *codec* untuk AVI.
2. MPEG (*Motion Picture Expert Group)*
   MPEG merupakan *file* terkompresi *lossy* yang biasanya digunakan untuk format VCD dengan *audio* berformat MPG. MPEG terdiri dari beberapa bagian:
   a. *Synchronization and multiplexing of video and audio*.
   b. *Compression codec for non-interlaced video signals*.
   c. *Compression codec for perceptual coding of audio signals*.
      MP1 or MPEG-1 Part 3 Layer 1 (MPEG-1 *Audio Layer* 1)
      MP2 or MPEG-1 Part 3 Layer 2 (MPEG-1 *Audio Layer* 2)
      MP3 or MPEG-1 Part 3 Layer 3 (MPEG-1 *Audio Layer* 3)
   d. *Procedures for testing conformance*.
   e. *Reference for testing conformace*.
   f. *Reference software*.
   MPEG-1 beresolusi 352 x 240 dan hanya men*support progressive scan video*. MPEG-2 digunakan untuk *broadcast*, siaran untuk *direct-satelit* dan *cable* tv. MPEG-2 *support interlaced* format. MPEG-2 digunakan dalam/pada *HDTV* dan DVD *vidoe disc*. MPEG-4 digunakan untuk *streaming*, CD *distribution*, *videophone* dan *broadcast television*. MPEG-4 mendukung digital *rights management*.
3. RMVB (*Real Media Variable Bitrate*)
   RMVB adalah sebuah format *video* digital yang dibuat oleh *RealNetworks, Inc,* yang memiliki kecepatan bit variabel perpanjangan dari *multimedia container RealMedia* format. RMVB biasanya digunakan untuk konten multimedia yang tersimpan secara lokal. *File* menggunakan format ini memiliki ekstensi *file*. RMVB.
   Kelebihan dari format RMVB adalah RMVB meninggalkan *Bit Rate* dan menggunakan *Variable Bit Rate* untuk kompres data video. File RMVB telah menjadi format populer untuk *video digital* karena mereka memiliki ukuran file yang lebih kecil dan kecepatan bit yang lebih rendah dengan kualitas yang lebih baik dibandingkan dengan AVI.
4. MKV (*Matroska Video*)
   MKV adalah salah satu format *video* yang mungkin sering dijumpai di internet. MKV merupakan alternatif format *video* selain beberapa format *video* digital seperti AVI,

MPEG, 3GP, RMVB dimana masing-masing memiliki sifat dan kualitas yang berlainan.

Format MKV biasanya digunakan untuk *video* dengan kualitas tinggi yang tidak semua PC mampu memutarnya dengan baik. Sebuah file *video* digital dalam format MKV memiliki beberapa bagian, yaitu :

a. *Video*
b. *Audio*
c. Subtitel

Semua bagian ini terpisah, namun menjadi satu bagian didalam format MKV. Bagian-bagian ini nantinya akan digabungkan menggunakan sebuah *codec* MKV sehingga *video* digital dalam format MKV ini dapat dibaca dan dijalankan menggunakan perangkat lunak multimedia player.

5. WMV (*Windows Media Video*)
   WMV adalah adalah format *file video* terkompresi yang dikembangkan oleh Microsoft. WMV, awalnya dirancang untuk aplikasi *Internet Streaming*, sebagai pesaing untuk *RealVideo*.
   File video digital dengan format WMV (*.wmv) menggunakan format pembawa ASF milik Microsoft. Berkas ini dapat dijalankan oleh perangkat lunak multimedia player seperti Windows Media *Player*, MPlayer, VLC media *player* atau Media *Player Classic*. Beberapa *player* pihak ketiga juga ada untuk berbagai *platform* seperti Linux yang menggunakan implementasi FFMPEG untuk *codec* WMV.
6. FLV (*Flash Video*)
   Flash *Video* (FLV) adalah *video* dengan format flash *movie* yang digunakan di Internet. FLV biasanya menjadi format standar yang digunakan oleh Youtube, Google *Video*, Reuters.com, Yahoo!Video, *MySpace*, dan lain-lain. *File* v*ideo* digital dengan format FLV biasanya ukurannya jauh lebih kecil daripada *video* digital yang menggunakan format MPEG atau AVI. Namun tentu saja kualitas dan resolusi *video* digital dengan *format* FLV lebih rendah darpada jenis *video* digital lainnya.
   Untuk memutar *file* dengan format FLV maka dibutuhkan sebuah *codec* khusus. Hal ini terjadi karena format FLV tidak bisa dimainkan dengan pemutar musik seperti *Winamp*, *Windows* Media *Player*, dll. Berbeda dengan format MPEG, AVI, 3gp, dan lain-lain, bisa dengan mudah dimainkan dengan pemutar musik tersebut.
7. 3GP (*Third Generation Project*)
   Format *video* 3GP merupakan format *video* untuk *mobile phone* dengan kompresi yang tinggi sehingga memiliki ukuran yang kecil namun dengan kualitas gambar yang cukup baik[1,6].

File 3GP adalah adalah bentuk simple atau ringkas dari format MPEG-4 Part 14 (MP4), yang dibuat untuk memperkecil besar dari ukuran *file* dan *bandwidth* sebuah telepon genggam. 3GP mempunyai 2 bentuk format standarnya yaitu :

a. 3GPP (*Third Generation Partnership Project*)
   Merupakan format 3GP yang digunakan untuk GSM-*based Phones*, dengan *file* name *extension* 3gp.
b. 3GPP2 (*Third Generation Partnership Project* 2)
   Merupakan format 3GP yang digunalan untuk CDMA-*based Phones*, dengan file name *extension* 3g2.

Format *video* 3gp sekarang banyak sekali digunakan, karena *file video* dengan format ini (3GP) memiliki ukuran yang kecil dan cocok sekali bila ingin menyimpan koleksi *video* pada perangkat *handphone*. Dan kebanyakan sekarang hampir disemua HP yang mempunyai fitur multimedia, menggunakan format *video* 3GP].

Steganografi adalah seni dan ilmu menulis atau menyembunyikan pesan tersembunyi dengan suatu cara sehingga selain si pengirim dan si penerima, tidak ada seorang yang mengetahui atau menyadari bahwa ada suatu pesan rahasia[2].

Sebaliknya, kriptografi menyamarkan arti dari suatu pesan, tapi tidak menyembunyikan bahwa ada suatu pesan. Kata steganografi (steganografi) berasal dari bahasa Yunani *steganos*, yang artinya “tersembunyi atau terselubung”, dan graphein, “menulis”.

Teknik steganografi meliputi banyak sekali metode komunikasi untuk menyembunyikan pesan rahasia (*teks* atau gambar) di dalam *file-file* lain yang mengandung *teks*, *image*, bahkan *audio* tanpa menunjukkan ciri-ciri perubahan yang nyata atau terlihat dalam kualitas dan struktur dari *file* semula. Metode ini termasuk tinta yang tidak tampak, *microdots*, pengaturan kata, tanda tangan digital, jalur tersembunyi dan komunikasi spektrum lebar. Tujuan dari steganografi adalah merahasiakan atau menyembunyikan keberadaan dari sebuah pesan tersembunyi atau sebuah informasi. Dalam prakteknya kebanyakan diselesaikan dengan membuat perubahan tipis terhadap data digital lain yang isinya tidak akan menarik perhatian dari penyerang potensial, sebagai contoh sebuah gambar yang terlihat tidak berbahaya. Perubahan ini bergantung pada kunci (sama pada kriptografi) dan pesan untuk disembunyikan. Orang yang menerima gambar kemudian dapat menyimpulkan informasi

ISSN : 2548-6985

terselubung dengan cara mengganti kunci yang benar ke dalam algoritma yang digunakan.

Pada metode steganografi cara ini sangat berguna jika digunakan pada cara steganografi komputer karena banyak format *file* digital yang dapat dijadikan media untuk menyembunyikan pesan. Format yang biasa digunakan diantaranya :

1. Format *image* : bitmap (bmp), gif, pcx, jpeg.
2. Format *audio* : wav, voc, mp3.
3. Format lain : *teks file*, html, pdf.

Kelebihan steganografi daripada kriptografi adalah pesan-pesannya tidak menarik perhatian orang lain. Pesan-pesan berkode dalam kriptografi yang tidak disembunyikan walaupun tidak dapat dipecahkan, akan menimbulkan kecurigaan. Seringkali steganografi dan kriptografi digunakan secara bersamaan untuk menjamin keamanan pesan rahasianya.

Sebuah pesan steganografi (*plaintext*), biasanya pertama-tama dienkripsikan dengan beberapa arti tradisional, yang menghasilkan *ciphertext*. Kemudian, *covertext* dimodifikasi dalam beberapa cara sehingga berisi *ciphertext*, yang menghasilkan *stegotext*. Contohnya, ukuran huruf, ukuran spasi, jenis huruf, atau karakteristik *covertext* lainnya dapat dimanipulasi untuk membawa pesan tersembunyi, hanya penerima (yang harus mengetahui teknik yang digunakan) dapat membuka pesan dan mendekripsikannya.

DCT (*Discrete Cosine Transform*)

Discrete Cosine Transform (DCT) biasa digunakan untuk mengubah sebuah sinyal menjadi komponen frekuensi dasarnya. DCT pertama kali diperkenalkan oleh Ahmed, Natarajan dan Rao pada tahun 1974 dalam makalahnya yang berjudul "*On image processing and a discrete cosine transform*" (Watson, 1994).

DCT mempunyai dua sifat utama untuk kompresi citra dan *video* yaitu :

1. Mengkonsentrasikan energi citra ke dalam sejumlah kecil koefisien (*energi compaction*).
2. Meminimalkan saling ketergantungan diantara koefisien-koefisien (*decorrelation*).

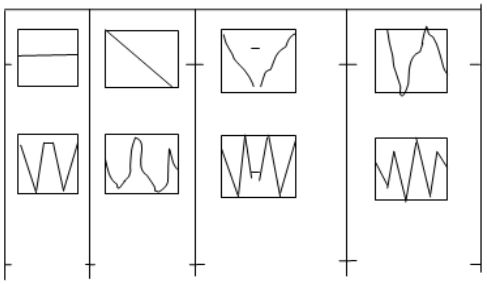
Discrete Cosine Transform dari sederet n bilangan real s(x), x = 0, … ,n-1, dirumuskan sebagai berikut :

$$S(u) = \sqrt{2/n}\, C(u) \sum_{x=0}^{n-1} S(x) \cos \frac{(2x+1)u\pi}{2n}$$

dengan u = 0,…, n - 1

dimana $C(u) = \begin{cases} 2^{-1}/_{2}, untuk\ u = 0 \\ 1\ \ , untuk\ lainnya \end{cases}$

Setiap elemen dari hasil transformasi S(u) merupakan hasil *dot product* atau *inner product* dari masukan s(x) dan basis vektor. Faktor konstanta dipilih sedemikian rupa sehingga basis vektornya *orthogonal* dan ternormalisasi. DCT juga dapat diperoleh dari produk vektor (masukan) dan n x n matriks *orthogonal* yang setiap barisnya merupakan basis vektor. π bernilai 180^0, Delapan basis vektor untuk n = 8, sebagaimana ditunjukkan pada Gambar 1, dimana setiap basis vektor berkorespondensi dengan kurva sinusoid frekuensi tertentu.



Gambar 1 Delapan Basis Vektor Untuk DCT Dengan n = 8

Barisan s(x) dapat diperoleh lagi dari hasil transformasinya S(u) dengan menggunakan invers discrete cosine transform (IDCT), yang dirumuskan sebagai berikut :

$$S(x) = \sqrt{2/n} \sum_{u=0}^{n-1} S(u)C(u) \cos \frac{(2x+1)u\pi}{2n}$$

dengan x = 0,…, n - 1

dimana $C(u) = \begin{cases} 2^{-1}/_{2}, untuk\ u = 0 \\ 1\ \ , untuk\ lainnya \end{cases}$

Persamaan di atas menyatakan s sebagai kombinasi linier dari basis vektor. Koefisien adalah elemen transformasi S, yang mencerminkan banyaknya setiap frekuensi yang ada di dalam masukan s (Watson, 1994).

Discrete Cosine Transform merepresentasikan sebuah citra dari penjumlahan sinusoida dari magnitude dan frekuensi yang berubah-ubah. Sifat dari DCT adalah mengubah informasi citra yang signifkan dikonsentrasikan hanya pada beberapa koefisien DCT. Oleh karena itu DCT sering digunakan untuk kompresi citra seperti pada JPEG.

METODE PENELITIAN

Pada dasarnya pengamanan pesan hanya menggunakan pola enkripsi dan depenelitian namun dalam penulisan penelitian ini pengamanan pesan dilakukan dengan cara

ISSN : 2548-6985

penyisipan ke dalam *video*, agar keamanan pesan tersebut dapat terjaga dari orang-orang atau pihak yang tidak yang bertanggung jawab. dengan cara menerapkan steganografi, implementasi penyisipan pada *video* dapat dilakukan dengan konsep penyisipan bit pesan dengan bit piksel *video* yang telah diubah ke dalam bentuk *frame*.

Steganografi *video* menggunakan metode DCT, DCT merupakan salah satu *transform coding* yang akan merubah *byte* data dari domain spasial menjadi domain frekuensi. Setelah *byte* diubah maka *byte* diubah ke bit dan selanjutnya dilakukan adalah penyisipan pesan dalam bilangan biner dengan bantuan algoritma LSB. Dimana penyisipan LSB dilakukan dengan menggantikan akhir bilangan biner DCT dengan bilangan biner pesan, hal ini dilakukan terus hingga semua pesan selesai disisipkan kebilangan biner DCT. Setelah penyisipan pesan selesai disisipkan, ubah bilangan biner hasil penyisipan menjadi bilangan decimal yang berguna unntuk melakukan *invers* DCT. *invers* DCT yang berfungsi mengembalikan bilangan biner yang telah di DCT sebelumnya.

Analisa Ekstrak *File Video*

Video di ektraksi sehingga terbagi ke dalam beberapa *frame* dan setiap *frame* dipisah ke dalam masing-masing segmen *grayscale*. Analisa untuk mendapatkan *frame* dari *video* digunakan aplikasi *convert video* to *jpeg*. Berikut contoh gambar *video* yang akan di ekstraksi dengan format 3GP *grayscale video* pada gambar 2 dibawah ini.

Gambar 2 file Video

Video di atas "videoabdi.3gp" berdurasi 13 detik, berukuran 506 KB, *video* akan di *convert* dengan aplikasi *convert video* to *jpeg* dan dimana pengestrakan dari aplikasi tersebut, pertama mengambil *video* format 3gp *grayscale*, pilih jumlah *frame*, pilih waktu perdetik dan jumlah *frame* dalam *video*. Hasil dari pengestrakan dapat dilihat pada gambar 3.

Gambar 3 Kumpulan Frame video

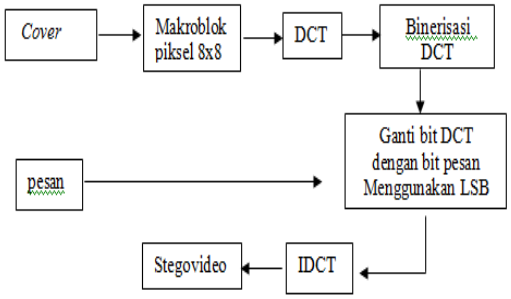
Dapat diketahui hasil ekstraksi memiliki 130 *frame file video*, dan 1 detik 10 *frame*. Dikarenakan pilihan dari aplikasi tersebut minimal 10 *frame* dan waktu minimal 1 detik. Dan masing-masing *frame video* berbentuk *frame grayscale* Dimana *frame* memiliki resolusi 192x144 piksel untuk setiap *frame*-nya. Setiap *frame* dalam *video* menjadi sebuah *file jpeg*.

Pada kapasitas penyimpanan pada setiap *frame* akan membagi piksel-piksel tersebut ke dalam blok-blok kecil dengan ukuran 8x8 sehingga setiap *frame video* tersebut memilki makroblok sejumlah 24 x 18 = 432 makroblok. Setiap piksel terdiri atas 1 *byte grayscale*. Setiap *frame* = 432 x 1 = 432 bit.
Untuk sebuah *video* = 13 x 432 = 5616 bit dan jumlah tersebut dapat menampung = 5616 / 8 = 702 karakter.

Analisa Penyisipan Pesan Pada *Video* Dengan Metode DCT

Proses penyisipan pesan rahasia yaitu bagaimana pesan rahasia disisipkan pada *video* dengan format 3gp sehingga pesan tersebut tidak diketahui keberadaanya. Pada proses ini, proses penyisipan membutuhkan masukan yaitu *cover video* sebagai tempat penyisipan pesan rahasia. Media *cover* yang digunakan adalah *video* format 3gp dan pesan rahasia yang disisipkan berupa teks maka penyisipan pesan dengan teks karakter yang ukurannya tidak melebihi daya tampung penyimpanan dari setiap *frame*. Metode penyisipan DCT ini adalah dengan memanfaatkan algoritma LSB dalam menyisipi pesan dengan cara mengganti bit ke 8, 16 dan 25 pada representasi biner *frame* dengan refresentasi pesan rahasia yang akan

disembunyikan. Aktifitas yang akan dilakukan pada proses penyisipan pesan ini adalah sebagai berikut :



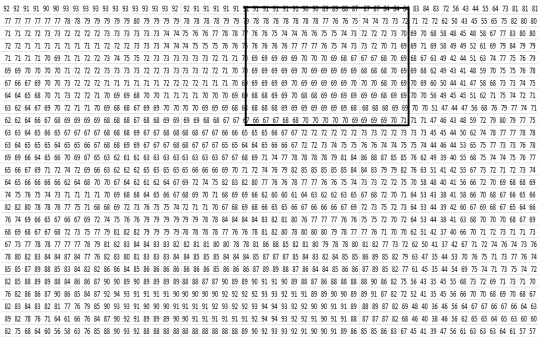
Gambar 4 Skema penyisipan pesan

Keterangan tahapan proses penyisipan pada gambar 4 dapat dijelaskan sebagai berikut :

1. Ekstraksi cover dan pesan
   Video diektraksi sehingga terbagi ke dalam beberapa *frame*, pada proses ini setiap informasi yang akan digunakan pada tahap selanjutnya akan disimpan. Dan pesan diekstrak ke dalam bentuk biner, digunakan pada tahap selanjutnya.
2. Makroblok piksel
   Setiap *frame* dari *video* akan dipecah menjadi makroblok ukuran 8x8,
3. DCT
   Menerapkan DCT terhadap makroblok-makroblok tersebut dengan rumus DCT 1D.
4. Binerisasi DCT
   Proses DCT pada makroblok akan menghasilkan bilangan matriks. Dan dilakukan binerisasi terhadap nilai sehingga dapat diperoleh bit LSB yang digunakan penyisipan.
5. Penyisipan biner pesan
   Proses penyisipan biner pesan yaitu penggantian biner LSB pada *cover* dengan bit-bit biner pesan.
6. Invers DCT
   Setelah pesan disisipkan, selanjutnya mengembalikan *cover* menjadi *video* 3gp seperti semula. Tahapan yang dilakukan merupakan kebalikan dari tahap penyisipan. Makroblok hasil *invers* DCT kemudian digabungkan kembali dengan maroblok yang lain sehinggan membentuk *stegovideo*.

Adapun contoh kasus proses penyisipan pesan rahasia tersebut yaitu sebagai berikut :

1. Misalnya *cover frame* pertama dari *video* dilakukan penyisipan pesan, dan nilai piksel dari *frame* tersebut. Untuk mencari nilai piksel dari *frame* dilakukan menggunakan matlab maka hasil dari pencarian nilai piksel dapat dilihat gambar 5 di bawah ini.



Gambar 5 Hasil Matriks dari frame

2. Matriks pada proses pertama kemudian di pecah menjadi makroblok dengan ukuran 8 x 8 sehingga 432 makroblok untuk setiap *frame*. Proses makroblok dilakukan untuk setiap *frame video*. Matriks hasil makroblok dari matriks dapat dilihat pada gambar 6.

92	92	91	91	90	90	93	93
77	77	77	77	77	77	78	78
71	71	72	72	73	73	72	72
72	72	71	71	71	71	71	71
71	71	71	71	70	69	71	72
69	69	70	70	70	70	71	72
67	66	67	69	70	70	73	72
64	64	65	68	70	71	73	72

Gambar 6 Makroblok 8 x 8 Pada Frame

3. Tahap selanjutnya proses DCT dilakukan terhadap makroblok yang akan digunakan sebagai penampung pesan. Maka proses DCT dapat digunakan dengan menggunakan rumus DCT 1D yaitu :

$$S(u) = \sqrt{2/n}\ C(u) \sum_{x=0}^{n-1} S(x) \cos \frac{(2x+1)u\pi}{2n}$$

dimana $C(u) = \begin{cases} 2^{-1}/_{2}, untuk\ u = 0 \\ 1 \quad , untuk\ lainnya \end{cases}$

Dari hasil perhitungan proses DCT di atas maka didapatkan hasil matriks pada makroblok dapat di lihat pada gambar 7.

183	-0	0	-0	0	0	-0	0
154	-5	-1	-5	0	2	-0	0
144	-1	-1	-3	0	-3	2	-0
142	-5	-1	-3	-0	0	-8	-0
141	-5	-1	-3	0	0	-28	0
140	3	-1	-3	0	1	-0	1
138	-6	-0	-3	0	-0	-0	0
137	-0	-1	-4	-0	-0	0	0

Gambar 7 Hasil DCT 1D pada makroblok

4. Kemudian makroblok yang telah dihasilkan dengan rumus DCT ubah bilangan decimal menjadi bilangan biner. Yang terlihat seperti dibawah ini :

ISSN : 2548-6985

11010111 10000000 00000000 10000000 00000000 00000000 10000000 00000000
10011010 10000101 10000001 10000101 00000000 00000010 10000000 00000000
10010000 10000001 10000001 10000011 00000000 10000011 00000010 10000000
10001110 10000101 10000001 10000011 10000000 00000000 10011100 10000000
10001101 10000101 10000001 10000011 00000000 00000000 10011100 00000000
10001100 00000011 10000001 10000011 00000000 00000001 10000000 00000001
10001010 10000110 10000000 10000011 00000000 10000000 10000000 00000000
10001001 10000000 10000001 10000100 10000000 10000000 00000000 00000000

Hasil DCT yang telah ubah kedalam bentuk bilangan biner. Selanjutnya dilakukan penyisipan pesan menggunakan bantuan algoritma LSB.

5. Setelah bit-bit biner tersebut di dapatkan proses selanjutnya menyisipkan pesan rahasia kedalam *frame* tersebut, pesan yang di sisipkan berupa teks dengan karakter ABDI, maka pesan tersebut akan dikonversikan ke dalam biner dapat di lihat dibawah ini.

A	0	1	0	0	0	0	0	1
B	0	1	0	0	0	0	1	0
D	0	1	0	0	0	1	0	0
I	0	1	0	0	1	0	0	1

Sisipkan bit per bit dari setiap karakter tersebut pada setiap makroblok *frame*, yang disisipkan berjumlah 32. Dalam proses penyisipan bit-bit tersebut DCT memanfaatkan algoritma LSB yaitu mengganti nilai bit paling kanan atau bit akhir dengan nilai-nilai bit data. Berikut di ilustrasikan bit pertama .

Biner DCT ← Bit pesan
11010111 ← 0

Maka biner hasil penyisipan 11010110.
Sehingga di dapat hasil penyisipan adalah sebagai berikut :

10110110 10000001 00000000 10000000 00000000 00000000 10000000 00000001
10011010 10000101 10000000 10000100 00000000 00000010 10000001 10100100
10010000 10000001 10000000 10000100 00000000 10000011 00010110 10000000
10001110 10000101 10000000 10000011 10000001 00000000 10011100 10000001
10001101 10000101 10000001 10000011 00000000 00000000 10011100 00000000
10001100 00000011 10000001 10000011 00000000 00000001 10000000 00000001
10001010 10000110 10000000 10000011 00000000 10000000 10000000 00000000
10001001 10000000 10000001 10000100 10000000 10000000 00000000 00000000

Setelah pesan disisipkan maka biner hasil dari penyisipan di kembalikan ke dalam bentuk desimal untuk dilakukan *invers* DCT. Dapat dilihat pada gambar 8.

182	-1	0	-0	0	0	-0	1
154	-5	-1	-5	0	2	-0	0
144	-1	-1	-3	0	-3	2	-0
142	-5	-1	-3	-0	0	-8	-0
141	-5	-1	-3	0	0	-28	0
140	3	-1	-3	0	1	-0	1
138	-6	-0	-3	0	-0	-0	0
137	-0	-1	-4	-0	-0	0	0

Gambar 8 Hasil penyisipan Bilangan Desimal

6. Kemudian nilai matriks tersebut di atas dikembalikan dari domain frekuensi ke domain spasial (ruang). Untuk melakukan hal itu, maka pada blok matriks ini dilakukan transformasi IDCT(*invers* DCT), dengan menggunakan rumus:

$$S(x) = \sqrt{2/n} \sum_{u=0}^{n-1} S(u)C(u) \cos \frac{(2x+1)u\pi}{2n}$$

dimana $C(u) = \begin{cases} 2^{-1}/2, untuk\ u = 0 \\ 1 \quad , untuk\ lainnya \end{cases}$

Maka didapatkan hasil dari perhitungan transformasi *invers* DCT dari rumus diatas tersebut dapat dilihat pada gambar 9.

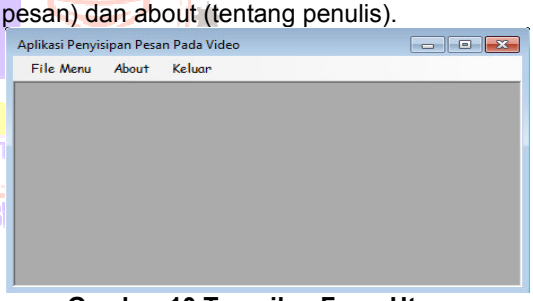
91	88	83	75	62	50	70	70
71	76	70	69	68	70	70	70
71	69	68	68	66	62	70	69
72	70	68	67	69	74	60	60
72	73	66	66	66	69	75	70
69	68	65	68	70	69	75	70
68	66	65	69	70	69	68	67
64	65	63	68	68	68	66	64

Gambar 9 Hasil *invers* DCT 1D

7. Tahap akhir dari proses penyisipan adalah pembentukan kembali *frame-frame* hasil penyisipan ke bentuk *video* 3gp sehingga menjadi bentuk *stegovideo*.

HASILDAN PEMBAHASAN

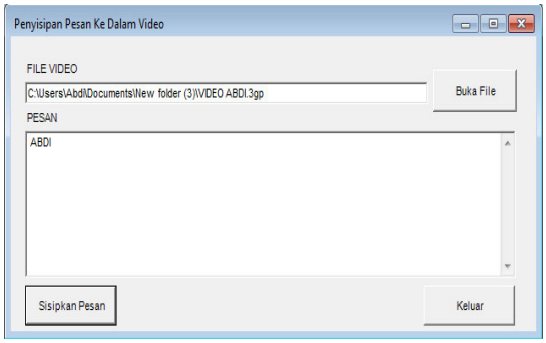
Gambar 10 berikut menunjukkan implementasi antarmuka pada perangkat lunak *SteganoVideo* saat memilih *mode Sembunyikan* (penyisipan pesan) dan about (tentang penulis).



Gambar 10 Tampilan Form Utama

Tampilan program di atas merupakan tampilan program utama, dalam *form* ini terdapat beberapa menu yaitu penyisipan pesan memanggil program penyisipan pesan, about memanggil tentang penulis. Dan keluar untuk keluar dari aplikasi.

Form berikut ini menunjukkan tampilam program penyisipan pesan pada *video* dan dapat dilihat pada gambar 11 berikut.



Gambar 11 Tampilan Program Penyisipan Pesan

Pada gambar di atas dimana pesan teks sudah berhasil tersisipi kedalam *video* tersebut. Penyisipan pesan di mulai dengan klik tombol buka *file* untuk menginputkan *file video* dengan format 3gp, untuk menampilkan pada kolom *file video* sebelum disisipi, kemudian isikan *teks* yang mau disisipkan pada kolom pesan, selanjutnya klik tombol sisipkan pesan untuk menyimpan pesan pada *video*. Dan tombol keluar untuk keluar dari aplikasi.

Form ini merupakan tampilan dari identitas penulis.

Gambar 12 Form About

KESIMPULAN

Dari hasil penulisan dan analisa sebelumnya, maka diambil kesimpulan-kesimpulan, sehingga penulisan ini dapat lebih bermanfaat. Adapun kesimpulan-kesimpulan tersebut adalah sebagai berikut :

1. Proses ekstraksi *file video* ke dalam bentuk *frame* dilakukan dengan aplikasi *convert video to jpeg* untuk mendapatkan *frame-frame* dan dilakukan penyisipan pesan pada *frame-frame* tersebut.
2. DCT merupakan salah satu *transform coding* yang akan merubah *byte* data dari domain spasial menjadi domain frekuensi.
3. Kualitas berkas *video* yang dihasilkan bergantung dari besarnya ukuran pesan pada perangkat lunak yang dibangun. Tujuan dari penggandaan ini adalah membuat agar pesan terlihat seperti *noise* pada berkas *video* 3gp sehingga pihak yang tidak berwenang tidak menyadari keberadaan pesan.

Penulis juga memiliki saran-saran yang dapat diberikan untuk pengembangan lebih lanjut kepada mahasiswa atau para pembaca. Adapun saran-saran yang dapat penulis berikan antara lain :

1. Perlu adanya dilakukan pengembangan untuk pengambilan pesan dari dalam *video.*
2. Perlu adanya analisis lebih lanjut dan implementasi teknik DCT pada format *video* lainnya seperti AVI, FLV, MOV, RMVB, GOM, MPEG.
3. Perlu dilakukan pengembangan format 3gp *grayscale* ke format 3gp berwarna.
4. Perlu dilakukan pengembangan untuk meningkatkan kapasitas penyisipan dalam arti penyisipan dapat dilakukan selain dalam *frame* dari berkas *video* 3gp.

DAFTAR PUSTAKA

1. Andres Nicolas tarigan, 2014. "*Pembuatan Aplikasi Penyisipan Pesan pada File Mp3*". Issn: 2301-9425, vol:III,nomor 2,2, Penerbit Pelita Informatika
2. Basuki Rrahmad, 2010 ."*Steganografi Menggunakan Metode Least Significant Bit Dengan Kombinasi Kriptografi*", volume 5, nomor 2, 6. Penerbit jurnal dinamika informatika
3. Elex media komputindo, 2010."*Membangun Aplikasi Database dengan Visual Basic 2008 dan SQL Server 2008*". Penerbit elex media komputindo
4. Jogiyanto H.M 2007 , "*Sistem Teknologi Informasi*", Penerbit Andi, Yogyakarta.
5. Rosa A.S dan M.Shalahuddin, 2011. " *Rekayasa Perangkat Lunak*", Bandung.
6. Tri Prasetyo Utomo, UIN Sunan Gunung Djati Bandung, 2013 ,"*Steganografi Gambar Untuk Proteksi Komunikasi pada Media Online*".
7. Syarifuddin, 2012 ."*Sistem Cerdas Deteksi citra dengan metode discrete cosine transform*", Universitas Hasanuddin, Makassar.
8. http://Digilib.unila.ac.id./3328/13. diakses pada pukul 01.05 tanggal 29 Mei 2015, penerapan.pdf)